# A REFERENCE ARCHITECTURE OF INTERNET OF THINGS ECOSYSTEM

**Stanimir Stoyanov, Asya Stoyanova-Doycheva, Todorksa Glushkova,**
**Emil Doychev, Jordan Todorov**
*University of Plovdiv „Paisii Hilendarski”*

***Abstract.*** *Summing up the experience of building Virtual Education Space, this paper presents a reference architecture known as Virtual Physical Space (ViPS). ViPS is being developed as a Cyber-Physical-Social-Space. Our goal is for the reference architecture to be adaptable for a variety of Internet of Things ecosystems in different domains such as smart cities, smart environment and agriculture, and intelligent medicine. This paper presents the first version of ViPS's general reference architecture and its core components.*

***Key words.*** *Cyber-Physical and Cyber-Physical-Social Spaces, Internet of Things, Intelligent Agents, Reference Architectures, Virtual Physical Space.*

## 1. INTRODUCTION

Within the DeLC project over the years, an environment with the same name has been built to support various forms of e-learning [1]. The environment is successfully used to support the educational process at the Faculty of Mathematics and Informatics of the University of Plovdiv and at the Faculty of Computer Science and Engineering of Burgas Free University. During a reengineering process, taking into account the various deficiencies of the environment – mainly the lack of opportunities for integrating the surrounding physical world, it was transformed into a Virtual Education Space (VES) [2]. At first glance, the rationale to integrate the physical world with the virtual environment for e-learning is not obvious; it is enough to point out that for disabled learners the physical world is of utmost importance in order to conduct a successful learning process.

Summing up the experience of building VES and taking into account the implementation level reached, as well as the inclusion of standards and formalisms, we believe that a new approach is needed to continue the work. The further development of the space will be divided in two separate projects. The first project will focus on creating a Reference Architecture for the Internet of Things Ecosystems, called ViPS (Virtual Physical Space), which can be adapted to various domains. The second project will continue the implementation of VES but already as an adaptation of the ViPS architecture for a particular field of application – e-learning.

This paper presents the first version of ViPS's general reference architecture and its core components. The rest of the paper is organized as follows: a short review of Cyber-Physical and Cyber-Physical-Social Spaces is considered in Section 2; this is followed by a brief overview of ViPS aspects in Section 3; Section 4 describes the ViPS architecture and its core components; finally, Section 5 concludes the paper.

## 2. CYBER-PHYSICAL VS. CYBER-PHYSICAL-SOCIAL SYSTEMS

In general, there are various concepts closely related to and competing with the notion of Cyber-Physical System (CPS), such as embedded systems, the Internet of Things (IoT), Industrial Internet, Internet of Machine-to-Machine, Industry 4.0, Intelligent Planet, Cyber-Physical-Human Systems (CPHS), Intelligent Systems, and Adaptive Systems [3]. Despite the fact that each of these areas has its supporters and communities, in [4] it is convincingly argued that the concept of CPS is fundamental because it embodies the basic engineering problem of integration of the digital and physical world. The concept of IoT is closely linked with that of CPS in recent studies, with disparate views on the similarities and distinctions between the two concepts. Some studies [5] assert that while CPS focus on bridging the physical and cyber worlds, IoT is concerned with the unique identification of heterogeneous devices and smart objects and their connectivity to the Internet. Some authors [6] state that while there are similarities between CPS and IoT, i.e., device cooperation to reach defined goals, IoT has a horizontal view comprised of hardware components interacting with each other, whereas CPS takes a vertical approach encompassing networked hardware, computational processes and control mechanisms. Other studies [7], while acknowledging the difference in system architectures of CPS and IoT, refer to them interchangeably. An increased level of abstraction in IoT is attributed to a large number of undefined factors, which are known only in terms of contribution to the system operation [8]. In particular, the concept of IoT is closely related to CPS. On the one hand CPS focuses on linking physical and cyber worlds; on the other hand IoT deals with the unique identification of heterogeneous devices and intelligent objects and their Internet connectivity. At the same time, while there are similarities between CPS and IoT – that is, device interoperability, IoT has a horizontal view consisting of interfacing hardware components while CPS adopts a vertical approach involving network hardware, computing processes, and control mechanisms.

One major component of the systems, however, is often overlooked or only partially addressed: the user [9]. However, there are significant implications for the user becoming part of the CPS. A new research domain involving human interactions is known as Cyber-Physical-Social Systems (CPSS) [10]. The CPSS integrates various data originating from physical, cybernetic and social spaces through synthesis techniques to provide human-readable abstractions and conclusions. CPSS applications in urban environments are geared toward reacting to the physical world and gaining knowledge of its state. The growing computer paradigm of CPSS is based on the technological development of CPS and Cyber-Social Systems (CSS) [11, 12]. This technological paradigm considers human and social dynamics as an integral part of CPS [13]. CPSS are characterized by a deep interaction between sensors, actuators and intelligent objects that are in the physical world, i.e. technology-rich social interactions and advanced reflections applied to collective intelligence [14].

Intelligent cities are typical of CPSS, which are possible through the introduction of cheap sensors, government initiatives on open public data, and citizens sharing and exchanging messages on urban networks on social networks. The vast amounts of data gained by the capture of physical phenomena through distributed sensing networks and those contributed by city residents through their sensory smartphones and online social networks can provide real-time large-scale data for intelligent urban services [15].

Retrieving knowledge from data, usually through large data analysis techniques, can help build a picture of urban dynamics that can enable intelligent applications and services and lead the decision-making process for both urban authorities and for the inhabitants of the city [16]. CPSS applications appear in the everyday life of intelligent urban systems in areas such as commands and controls, intelligent environments, intelligent transport, intelligent social manufacturing systems, and so on. Such applications rely on effective monitoring of urban physical infrastructure and the environment and combine collected data through intelligent cyber processes to provide better services to citizens; for example by adjusting the traffic signal time based on vehicle arrival information and cyclists [17], development of sustainable waste management systems [18] and recommendations for events or locations based on citizens' preferences, proximity, pathways and environmental conditions (eg pollution levels) [19]. The major urban information system offers the potential for creating more sustainable and environmentally friendly future cities [20].

## 3. ASPECTS OF VIPS IN A NUTSHELL

To motivate the proposed ViPS architecture, it is necessary to mention the following three general aspects. The first one addresses the **Personal Assistants** (PAs). ViPS, developed as a CPSS ecosystem, puts the Personal Assistants (PAs) in the spotlight. Because the PAs operate as rational BDI agents [21] all their mental states are represented by events. Therefore, events are fundamental to the operation of the ViPS.

The second point is that ViPS as a reference architecture for IoT ecosystems has to provide opportunities for virtualization of real „things” taking into account **time**, **space** and **events**. In this way, it is essential to propose appropriate formalisms for presentation and work with the temporal, spatial and event aspects of things of interest.

The next point is that ViPS is supported by active components known as **Operative Assistants** (OAs) implemented also as rational BDI agents. The OEs are architectural components suitable for providing the necessary context-awareness, reactivity and proactivity of the space. However, they are unsuitable to deliver the business functionality provided in the space. For this reason, OAs interact closely with **services** or **micro-services**.

The last aspect concerns the **integration** between the virtual world and the physical world. We prefer to do this through a transparent intelligent layer of intelligent agents known as **guards**. The guards carry the data transfer from the physical world to the virtual space as well as information in the opposite direction from the virtual to the physical world.

## 4. VIPS ARCHITECTURE

The proposed ViPS architecture is presented in Fig.1. In this section the basic components of the ViPS are presented briefly.

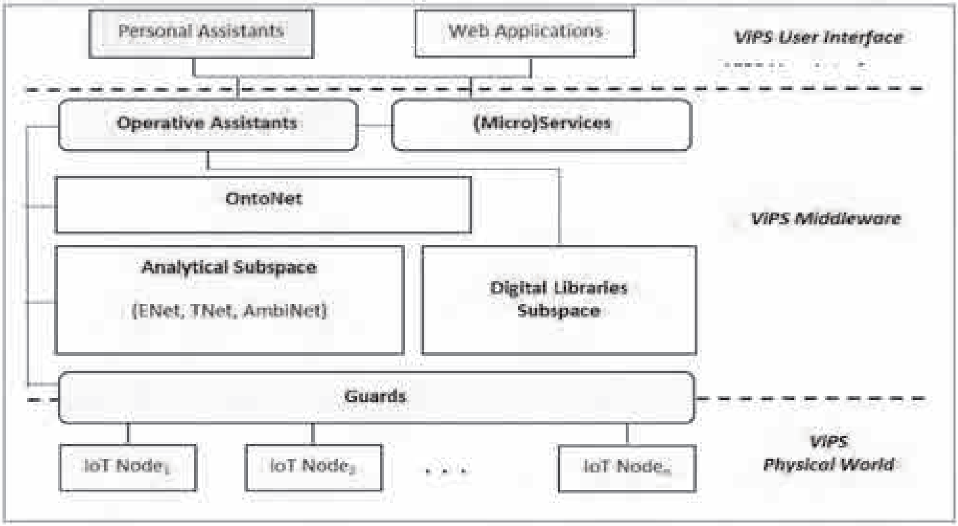

*Fig. 1. ViPS Architecture*

**Personal Assistants.** As mentioned, the PAs are closely related to the events; in this way the event model is a first-class citizen in the space. Why does the event model have a fundamental meaning for ViPS? The life cycle of a PA can be presented as a sequence of actions to „complete“ various events related to the management of the user's „Personal Schedule“ (PS). The PS is an event-based structure, i.e. its entry points are domain-events which in turn are indexed by base or system events (Fig. 2.). Events can be used for specification of policies presenting operative scenarios of the space.

As a rational agent with a BDI-like architecture, the PA consists of two basic components – Deliberation and Means-Ends Reasoning. In the deliberation phase, in order to determine the PA’s actual goal, the PS is mapped to mental states of the PA in the following way:

- Domain-events represent the Desires of the PA;
- The PS’s indices ([date, time, location]) are Beliefs of the PA;
- *ε* is used as a kind of metric to determine the PA's Intention.

The PS is a structure consisting of the elements *<(event, type) [date, time, location] ; (type, ε)>* where:

- *event* – is an event identifier;
- *type* – an event type, usually a domain-event (in accordance with the Event Model);
- *[date, time, location]* – an index consisting of the basic events date, time, location (in accordance with the Event Model;
- *ε – epsilon* event interval, i.e. an interval of interest preceding the occurrence of the event *event*. In this interval, the PA normally performs preventive work. Essentially, *ε* guides the search in the PS by helping to identify the event (events) entering in the PA's vision (control).

Furthermore:

- If *time* is a "significant" index then the PA interacts with the TNet;
- If *location* is a "significant" index then the PA interacts with the AmbiNet;
- If both are "significant," then the PA interacts with both networks.

*Fig. 2. PS Structure*

One of the basic principles on which the PA is built is that of „anticipating action“. The PA has an early warning system, i.e. it has to be able to perform certain warnings before the event occurs. In the timeframe before the event occurs, the PA cannot rely on the Event Engine (EE); the EE is triggered when an event occurs. To ensure the implementation of the forerunner action strategy, we propose the interaction between the PA and the AmbiNet, respectively TNet.

Theoretically, the life cycle of the PA can be defined as management of the interval $I(\Delta + T_E - \varepsilon)$ where:

- $T_E$ is the time of the event $E$;
- $\varepsilon$ is an interval preceding $T_E$;
- $\Delta$ is the duration of the event $T_E$.

$\varepsilon$ and $\Delta$ may vary within a range depending on the current domain. Also $\varepsilon = \Delta = 0$ is conceivable; e.g., $\Delta = 0$ in instant events or $\varepsilon = 0$ if the interval preceding the event does not concern the PA, i.e. it does not affect the prevention or anticipatory actions. For example, in e-learning there may be an agreement according to which different events will be different – e.g. exam $\varepsilon = 2$ weeks, lecture $\varepsilon = 2$ days, exercise $\varepsilon = 3$ days.

Our idea is for the interval $\varepsilon$ is to be managed by CCA and AjTempura. As regards AjTempura:

- To $\varepsilon$ an ITL interval is mapped and in this way AjTempura can be involved in the operating of ViPS;
- The necessary space activities during this interval are managed by AjTempura, i. e. AjTempura becomes proactive (it is implemented as a multi-agent system).

**Analytical Subspace.** As mentioned previously, for virtualization of things, time and space aspects, as well as events, they have to be accepted. That is why one of the most interesting, yet the most sophisticated components of ViPS, is the analytical subspace aiming to provide opportunities for time, space and events to work in a formal way. In this subspace, three interacting formal systems are located: ENet, AmbiNet and TNet.

*ENet* provides a formal presentation of the individual events eligible for the domain of interest. Furthermore, links between events can be also specified in this network, according to suitable classifications. ENet is constructed in accordance with the Event Model presented in more detail in [22]. We believe that the event model could be the major mechanism for synchronizing the performance of the space components. Events can happen in any place and at any time in the ViPS. When events occur the Event Engine uses ENet to correctly identify, interpret and manage the particular event. The Event Engine itself has been implemented as a special OA. A next tool supporting the ENet is the Event Editor that can be used to create a hierarchy (network) of events relevant to the particular main one.

*AmbiNet* is used to formally present the spatial aspects (incl. location) of the „things” virtualized in the space. For this purpose, we use an approach based on the concept of „ambient“ and ambient-oriented modeling, respectively. The specific formalism used in ViPS is Calculus of Context-aware Ambients (CCA) [23]. CCA provides appropriate mathematical notation and tools for modeling mobile and context-aware systems as ambients. In [24], a context-aware model of DeLC is presented in detail, and ideas and results of modeling of various aspects and components of VES are discussed in [25]. In fact, AmbiNet is a modeling environment, the kernel of which is the CCA interpreter implemented as a special operational assistant. In addition, a specialized editor (AmbiNet Editor) is developed to help create, update, and remove patterns in AmbiNet. Furthermore, intelligent search tools and a generator (Route Generator and Optimizer) of optimal thematic routes taking into account the wishes and preferences of users are also included in the network. For attractive presentation of the results gained by interpretation of the models, a tool (AmbiNet Visualizer) using virtual reality technologies is developed.

*TNet* is designed to represent the temporal aspects of the virtualized „things“ in a formal manner. The network is implemented according to the ITL specification [26] defining a temporal logic. The core of the TNet is the interpreter AjTempura implemented as a multi-agent specialized OA [27]. The basic principle of the interpreter is the processing of text strings representing ITL statements or formulas. Thus, albeit working correctly based on the mathematical model of ITL, it is difficult to apply directly to real systems. For this reason, an accompanying module (similar to AnaTempura [28]) has to be developed; it will operate as an interpreter's environment collecting and delivering information to be interpreted.

*ENet, AmbiNet and TNet interaction.* Managing the collaboration between these systems is a real challenge. For this purpose, an InterAction Protocol (IAP) is developed to take into account the nature, purpose and agent-oriented architecture of the three systems. In general, IAP has to conduct the following principle interaction scheme:

- The PA identifies the beginning of the interval $\varepsilon$ ;
- The PA interacts with AmbiNet, for example, an appropriate route to be generated;
- The PA interacts with AjTempura, for example, the generated route to be implemented where an interaction between AmbiNet and AjTempura is possible;
- In time $T_E$ ENet will be also activated (proactively) .

**Digital Libraries Subspace.** The objective of building a Digital Libraries Subspace (DLS) is for it to be a repository of background knowledge for various domains which ViPS is adapted for. The background knowledge of individual applications is structured and stored in separate, independent segments known as libraries. In this way, we believe the adapting of the reference architecture will be greatly facilitated. Over time, base versions of the libraries can be integrated as an integral part of the ViPS. Once adapted, the DLS has to provide opportunities to extend background knowledge for the particular domain. As concerns e-learning, the library may include e-learning content, test questions, publications, diploma works, and course projects. The libraries are served by specialized operational assistants, realized as BDI rational agents.

What is significant for this subspace is that it includes heterogeneous data repositories of different formats. Therefore, it is appropriate to build the subspace on two levels:

- Standard library – this is a standardized repository and supporting assistants independent of the domains to be adapted;
- Domain libraries – they provide knowledge for the domain to be adapted.

For example, using this approach, the current SCORM 2004 Engine can be transformed into a specialized OA that will operate in the standard library. For various applications, domain libraries delivering SCORM 2004 compliant content can be developed. Both libraries work together to interpret the learning content. Similarly, the electronic test module implemented as an operational assistant can be located in the standard library while tests and test questions are located in the domain libraries.

The standard library can also include a single common meta-level catalogue that provides intelligent search capabilities and supports the use of domain libraries. It would be expedient to develop the catalogue as an ontology or hierarchy (or network) of ontologies.

Our experience shows that the DSpace standard [29] provides powerful tools for building digital libraries. The standard is compliant with the Dublin Core specification [30] and at the same time it can be expanded depending on the domain of interest.

**OntoNet.** OntoNet is a hierarchical network of ontologies to support the work of the two subspaces. Our goal in building this network is to increase the intelligence of space using semantic modeling technologies. The network is constructed and managed as a matrix structure, i. e. horizontal and vertical levels are distinguished. Horizontally, it is structured in two levels – standard and applied. The standard level includes ontologies needed for the ViPS to operate regardless of the particular application. The applied level includes ontologies presenting knowledge about a specific domain, e.g., e-learning, cultural and historical heritage, smart city. Vertically, the OntoNet consists of an object level and a meta-level. At the object level, presentations of the application-specific content information are saved while the meta-level provides additional data known as meta-data supporting effective work with the object level.

Access to OntoNet resources is through specialized OAs.

**Guards.** The system of guards is responsible for the interaction between the virtual and the physical worlds. The guards carry the data transfer from the physical world, represented as a multitude of the so-called IoT Nodes, to the two virtual subspaces, as well as the data transfer in the opposite direction from the virtual world to the physical one. The IoT Nodes are configurations that typically include various types of sensors, controllers, and actuators located in dynamically constructed architectural layers. Typically, the controllers are single-chip or single-board computers (e.g., Raspberry Pi). The communication between a guard and IoT Nodes is usually performed through the Internet and free personal network technology (e.g., LoRa technologies).

A guard is able to collect data from a group of IoT nodes and to transmit it to the higher levels of space; the actual information from the physical world is delivered and stored in both subspaces. Conversely, guards can transmit commands to IoT Nodes to influence the physical world through their own actuators. Depending on the purpose, the guards may be of various types, e.g. biometrics, cybersecurity, and hazards such as floods, fires, or earthquakes. Another classification essential for the ViPS is based on the place where the guards are operating, i.e. in the physical or in the virtual world.

The architecture of the guard system is developed in accordance with the standardized framework proposed by Cisco, and it includes the two architectural stacks – the Core IoT Functional Stack and the IoT Data Management and Compute Stack [31]. The active components (guards) are implemented as intelligent rational agents that can be located at different levels mainly in the fog layer.

Within the proposed architecture, various types of guards can be developed. In addition to the already presented ones, in the ViPS there can be implemented guards providing effective and secure space operation such as responsiveness to humidity, temperature, oddness, or earthquake. The next types of guards under development are biometrics and cyber security guards. We are also considering other types of guards.

**Support of the C3A model.** For efficient use of ViPS resources, a context-aware-oriented agent architecture model known as C3A has been proposed [32]. This model can be implemented in the components of the ViPS as well.

## 5. CONCLUSION

For years, VES has been used to support the educational process at the Faculty of Mathematics and Informatics of the University of Plovdiv. The space is supported by various types of components. The active components are personal and operative assistants implemented as rational BDI agents. Since agents are not suited to provide functionality, they are combined with services including micro-services.

Currently, a new more abstract reference architecture known as ViPS is being designed that could be adapted for various IoT applications. The ViPS is implemented by the help of the development environments JADE [33] and Jadex [34].

**Acknowledgments.** The research is partly supported by the NPD – University of Plovdiv under Grant No. MU17-FMI-001 „EXPERT (Experimental Personal Robots That Learn)", 2017-18 and Grant No. FP17-FMI-008 „Innovative software tools and technologies with application in research in mathematics, informatics and pedagogy of education", 2017-18.

## REFERENCES

[1] S. Stoyanov, I. Popchev, E. Doychev, D. Mitev, V. Valkanov, A. Stoyanova-Doycheva, V. Valkanova , I. Minov, DeLC Educational Portal, Cybernetics and Information Technologies (CIT), Vol.10, No 3., Bulgarian Academy of Sciences, 2010, pp. 49-69.

[2] С. Стоянов, И. Попчев, Инфраструктури за електронно обучение, списание „Техносфера“, БАН, 4(30)/2015, ISSN 1313-38612015, 38-45.

[3] H. Müller, The Rise of Intelligent Cyber-Physical Systems, Computer, Dezember 2017, 7-9.

[4] E.A. Lee, The Past, Present and Future of Cyber-Physical Systems: A Focus on Models, Sensors, vol. 15, no. 3, 2015, pp. 4837–4869.

[5] Salim, F.; Haque, U. Urban computing in the wild: A survey on large scale participation and citizen engagement with ubiquitous computing, cyber physical systems, and Internet of Things. Int. J. Hum. Comput. Stud. **2015**, 81, 31–48.

[6] Bordel, B.; Alcarria, R.; Robles, T.; Martín, D. Cyber–physical systems: Extending pervasive sensing from control theory to the Internet of Things. Perv. Mob. Comput. **2017**, 40, 156–184.

[7] Shih, C.S.; Chou, J.J.; Reijers, N.; Kuo, T.W. Designing CPS/IoT applications for smart buildings and cities, IET Cyber Phys. Syst. Theory Appl., 2016, 1, 3–12.

[8] Hehenberger, P.; Vogel-Heuser, B.; Bradley, D.; Eynard, B.; Tomiyama, T.; Achiche, S. Design, modelling, simulation and integration of cyber physical systems: Methods and applications. Comput. Ind. **2016**, 82, 273-289.

[9] F. Dressler, Cyber Physical Social Systems: Towards Deeply Integrated Hybridized Systems, 2018 International Conference on Computing, Networking and Communications (ICNC), 2018, 420-424.

[10] A. Sheth, P. Anantharam, and C. Henson, „Physical-Cyber-Social Computing: An Early 21st Century Approach," IEEE Intelligent Systems, vol. 28, no. 1, Feb. 2013, 1541-1672.

[11] Zeng, J.; Yang, L.T.; Lin, M.; Ning, H.; Ma, J. A survey: Cyber-physical-social systems and their system-level design methodology. Future Gener. Comput. Syst. 2016.

[12] Ning, H.; Liu, H.; Ma, J.; Yang, L.T.; Huang, R. Cybermatics: Cyber–physical–social-thinking hyperspace based science and technology. Future Gener. Comput. Syst. 2016, 56, 504–522.

[13] Wang, F.Y. The Emergence of Intelligent Enterprises: From CPS to CPSS. IEEE Intell. Syst. 2010, 25, 85–88.

[14] S. De, Y. Zhou, I. L. Abad, K. Moessner, Cyber–Physical–Social Frameworks for Urban Big Data Systems: A Survey, Applied Sciences, October 2017.

[15] Guo, B., Wang, Z., Yu, Z., Wang, Y., Yen, N.Y., Huang, R., Zhou, X., Mobile Crowd Sensing and Computing: The Review of an Emerging Human-Powered Sensing Paradigm, ACM Comput. Surv., 2015, 48, 1–31.

[16] Zhou, Y., De, S.,Moessner, K., RealWorld City Event Extraction from Twitter Data Streams, Procedia Comput. Sci. 2016, 98, 443-448.

[17] Guo, W., Zhang, Y., Li, L., The integration of CPS, CPSS, and ITS: A focus on data, Tsinghua Sci. Technol. 2015, 20, 327–335.

[18] Digiesi, S., Facchini, F., Mossa, G., Mummolo, G., Verriello, R., A Cyber—Based DSS for a Low Carbon Integrated Waste Management System in a Smart City, IFAC PapersOnLine, 2015, 48, 2356–2361.

[19] Costanzo, A., Faro, A., Giordano, D., Spampinato, C., An ontological ubiquitous city information platform provided with Cyber-Physical-Social-Systems, In: Proceedings of the 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2016, 137–144.

[20] Guo, B., Yu, Z., Zhou, X. A, Data-Centric Framework for Cyber-Physical-Social Systems, IT Prof. 2015, 17, 4–7.

[21] M. Wooldridge, An Introduction to MultiAgent Systems, Wiley, 2009.

[22] Гуглев, Ж., Стоянов, С. Модел за представяне и разпространяване на събития във Виртуалното Образователно Пространство, Международна конференция „Дж. Атанасов, София, 2017, 261-264.

[23] Siewe, F., Zedan, H., Cau, A. The Calculus of Context-aware Ambients, Journal of Computer and System Sciences, 2011, 77, 597–620

[24] Al-Sammarraie, M., H. Policy-based Approach For Context-aware Systems, PhD Thesis, Software Technology Research Laboratory De Montfort University, 2011, Leicester, United Kingdom

[25] Стоянов, С., Глушкова, Т., Попчев, И. Моделиране на интелигентни контекстно-зависими системи, Инженерни науки, 3/2017, LIV, БАН, 2017, 5-21.

[26] Moszkowski B. Compositional reasoning using Interval Temporal Logic and Tempura, *Lect. Notes in Comp. Sci.,*1536, 1998, Springer, 439-464.

[27] Вълканов, В. Контекстно-ориентирано управление на електронни услуги, дисертация, Академично издателство „Проф. Марин Дринов“, 2013.

[28] AnaTempura, http://www.antonio-cau.co.uk/ITL/itlhomepagese6.html

[29] DSpace, https://duraspace.org/dspace/.

[30] Dublin Core Metadata Initiative, http://dublincore.org/.

[31] D. Hanes, G. Salgueiro, P. Grossetete, R. Barton, J. Henry, IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things, Cisco Press, USA, 2017.

[32] S.Stoyanov, V. Valkanov, I. Popchev, A. Stoyanova-Doycheva, E. Doychev, A Model of Context-Aware Agent Architecture, Compt. Rend. Acad. Bulg. Sci., Tome 67, № 4, 2014, pp. 487-496.

[33] JADE, http://jade.tilab.com/download/add-ons/.

[34] JADEX, https://www.activecomponents.org/.